

On the Valid Deterministic Localization Plan Problem

Andrey Sheka

Department of Intelligent Systems and Robotics
Ural Federal University
620083 Ekaterinburg, Russia
Andrey.Sheka@gmail.com

Copyright © 2013 Andrey Sheka. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract

Grid graphs are popular testbeds for planning with incomplete information. In particular, it is studied a fundamental planning problem, localization, to investigate whether gridworlds make good testbeds for planning with incomplete information. It is found empirically that greedy planning methods that interleave planning and plan execution can localize robots very quickly on random gridworlds or mazes. Thus, they may not provide adequately challenging testbeds. On the other hand, it is showed that finding localization plans that are within a log factor of optimal is NP-hard. Thus there are instances of gridworlds on which all greedy planning methods perform very poorly. These theoretical results help empirical researchers to select appropriate planning methods for planning with incomplete information as well as testbeds to demonstrate them. However, for practical application of difficult instances we need a method for their fast decision. In this paper, we consider an approach to solve localization problem. In particular, we consider an explicit polynomial reduction from the decision version of the valid deterministic localization plan problem to the 3-satisfiability problem.

Mathematics Subject Classification: 68T40

Keywords: 3-satisfiability problem, localization plan, grid graph, NP-complete

1 Introduction

Different gridworlds are popular testbeds for planning with incomplete information. A fundamental planning problem, localization, was studied in [1] to investigate whether gridworlds make good testbeds for planning with incomplete information. In particular, the valid deterministic localization plan problem (VDLPP) was proposed in [1]. The authors of [1] for VDLPP found empirically that greedy planning methods that interleave planning and plan execution can localize robots very quickly on random gridworlds or mazes. Thus, random gridworlds and mazes may not provide adequately challenging testbeds. On the other hand, the authors of [1] showed that finding localization plans that are within a log factor of optimal is **NP**-hard. Thus there are instances of gridworlds on which all greedy planning methods perform very poorly. In particular, the authors of [1] showed how to construct them. It is clear that these theoretical results help empirical researchers to select appropriate planning methods for planning with incomplete information as well as testbeds to demonstrate them. However, for practical application of difficult instances we need a method for their fast decision. Some practical applications of VDLPP were considered in [2]. Also, the authors of [2] were proposed to use an explicit polynomial reduction from VDLPP to satisfiability problems to obtain solutions for difficult instances of VDLPP. In this paper, we consider an explicit polynomial reduction from VDLPP to the 3-satisfiability problem (3SAT). Also, we present experimental results for the problem.

We assume that a grid graph G is given by a matrix

$$(g[i, j])_{m \times n}$$

where $0 \leq i \leq m - 1$, $0 \leq j \leq n - 1$, m and n are dimensions of G . Note that we can also assume that $g[i, j] = 1$ or $g[i, j] = 0$. In this case, $g[i, j] = 1$ if and only if the cell with coordinates i and j belongs to G . Following [1], a localization plan is valid if and only if there is no matter which cell the robot is started in, it eventually prints out its current cell or correctly determines that localization is impossible.

THE VALID DETERMINISTIC LOCALIZATION PLAN PROBLEM (VDLPP):

INSTANCE: *A grid graph G , a natural number K .*

QUESTION: *Is there a valid deterministic localization plan such that the worst-case performance of this plan does not exceed K ?*

Let H is a grid graph. Assume that H is given by a matrix

$$(h[i, j])_{m+2K \times n+2K}$$

such that

$$-K \leq i \leq K + m - 1;$$

$$-K \leq j \leq K + n - 1;$$

$$h[i, j] = 0 \text{ where } i \leq -1;$$

$$h[i, j] = 0 \text{ where } i \geq m;$$

$$h[i, j] = 0 \text{ where } j \geq m;$$

$$h[i, j] = 0 \text{ where } j \geq n;$$

$$h[i, j] = g[i, j] \text{ where } 0 \leq i \leq m - 1 \text{ and } 0 \leq j \leq n - 1.$$

Note that if we have a valid deterministic localization plan such that the worst-case performance of this plan does not exceed K then we have some sequence of instructions. We can assume that these instructions are defined as follows:

$$\begin{aligned} (0, 0) & \text{ "stop"}; \\ (1, 0) & \text{ "east"}; \\ (-1, 0) & \text{ "west"}; \\ (0, 1) & \text{ "north"}; \\ (0, -1) & \text{ "south"}. \end{aligned}$$

So, we can consider following sequence of instructions

$$(i[1], j[1]), \dots, (i[K], j[K])$$

such that

$$i[p] \in \{-1, 0, 1\} \text{ where } 1 \leq p \leq K;$$

$$j[p] \in \{-1, 0, 1\} \text{ where } 1 \leq p \leq K;$$

$$\wedge_{p=1}^K ((i[p] = -1 \rightarrow j[p] = 0) \wedge$$

$$(i[p] = 1 \rightarrow j[p] = 0) \wedge$$

$$(j[p] = -1 \rightarrow i[p] = 0) \wedge$$

$$(j[p] = 1 \rightarrow i[p] = 0)).$$

It is easy to check that the sequence of instructions give us a valid deterministic localization plan if and only if there is a sequence

$$(u(1), v(1), \dots, u(K+1), v(K+1))$$

such that

$$0 \leq u[q] \leq m-1 \text{ where } 1 \leq q \leq K+1;$$

$$0 \leq v[q] \leq m-1 \text{ where } 1 \leq q \leq K+1;$$

$$(\wedge_{q=1}^{K+1} g[u[q], v[q]] = 1) \wedge$$

$$(\wedge_{p=1}^K u[p] + i[p] = u[p+1] \wedge$$

$$v[p] + j[p] = v[p+1]) \wedge$$

$$(\wedge_{d=0}^{m-1} \wedge_{e=0}^{n-1} (\wedge_{p=0}^K (s[d, p] + i[p] = s[d, p+1] \wedge$$

$$t[e, p] + j[p] = t[e, p+1])) \wedge s[d, 1] = d \wedge t[e, 1] = e \wedge$$

$$(s[d, 1] \neq u[1] \vee t[e, 1] \neq v[1] \vee$$

$$(\vee_{q=1}^{K-1} h[s[d, q], t[e, q]] = 0))).$$

Using these considerations, it is not very difficult to construct a Boolean function which is true if and only if there is a valid deterministic localization plan consisting of K actions. Note that this function can be constructed so that, using a set of values on which the function is true, we automatically obtain a sequence of instructions. For instance, we can assume that

$$i[p] = -1 \Leftrightarrow (x[1, p] = 1 \wedge x[2, p] = 0);$$

$$i[p] = 1 \Leftrightarrow (x[1, p] = 0 \wedge x[2, p] = 1);$$

$$i[p] = 0 \Leftrightarrow (x[1, p] = 0 \wedge x[2, p] = 0);$$

$$j[p] = -1 \Leftrightarrow (y[1, p] = 1 \wedge y[2, p] = 0);$$

$$j[p] = 1 \Leftrightarrow (y[1, p] = 0 \wedge y[2, p] = 1);$$

$$j[p] = 0 \Leftrightarrow (y[1, p] = 0 \wedge y[2, p] = 0);$$

$$u[q] = a \Rightarrow u[a, q] = 1;$$

$$u[q] = a \Rightarrow u[b, q] = 0 \text{ where } b \neq a;$$

$$v[q] = a \Rightarrow v[a, q] = 1;$$

$$v[q] = a \Rightarrow v[b, q] = 0 \text{ where } b \neq a;$$

$$h[i, j] = 1 \Rightarrow z[a[-K], \dots, a[K + m - 1],$$

$$b[-K], \dots, b[K + n - 1]] = 1 \text{ where}$$

$$a[i] = b[j] = 1,$$

$$a[p] = b[q] = 0 \text{ for any } p \neq i, q \neq j;$$

$$h[i, j] = 0 \Rightarrow z[a[-K], \dots, a[K + m - 1],$$

$$b[-K], \dots, b[K + n - 1]] = 0 \text{ where } a[i] = b[j] = 1,$$

$$a[p] = b[q] = 0 \text{ for any } p \neq i, q \neq j.$$

In this case we can consider the following Boolean function:

$$(\wedge_{p=1}^K (\neg x[1, p] \vee \neg x[2, p])) \wedge$$

$$(\wedge_{p=1}^K (\neg y[1, p] \vee \neg y[2, p])) \wedge$$

$$(\wedge_{p=1}^K (\neg x[2, p] \rightarrow (\neg y[1, p] \wedge \neg y[2, p]))) \wedge$$

$$(\wedge_{p=1}^K (\neg x[1, p] \rightarrow (\neg y[1, p] \wedge \neg y[2, p])) \wedge$$

$$(\wedge_{p=1}^K (\neg y[2, p] \rightarrow (\neg x[1, p] \wedge \neg x[2, p])) \wedge$$

$$(\wedge_{p=1}^K (\neg y[1, p] \rightarrow (\neg x[1, p] \wedge \neg x[2, p])) \wedge$$

$$(\wedge_{q=1}^{K+1} (\vee_{p=1}^m u[p, q])) \wedge$$

$$(\wedge_{q=1}^{K+1} (\vee_{p[1] \neq p[2]} (\neg u[p[1], q] \vee \neg u[p[2], q]))) \wedge$$

$$1 \leq p[1] \leq m$$

$$1 \leq p[2] \leq m$$

$$(\wedge_{q=1}^{K+1} (\vee_{p=1}^n v[p, q])) \wedge$$

$$(\wedge_{q=1}^{K+1} (\vee_{p[1] \neq p[2]} (\neg v[p[1], q] \vee \neg v[p[2], q]))) \wedge$$

$$1 \leq p[1] \leq n$$

$$1 \leq p[2] \leq n$$

$$(\wedge_{a[i] \in \{0,1\}; \quad z[a[-K], \dots, a[K+m-1],$$

$$b[j] \in \{0,1\};$$

$$-K \leq i \leq K+m-1;$$

$$-K \leq j \leq K+n-1;$$

$$a[l]=b[r]=1;$$

$$h[l, r]=1;$$

$$a[p]=0, p \neq l;$$

$$-K \leq l \leq K+m-1;$$

$$b[q]=0, q \neq r;$$

$$-K \leq q \leq K+n-1$$

$$b[-K], \dots, b[K+n-1])) \wedge$$

$$(\wedge_{a[i] \in \{0,1\}; \quad \neg z[a[-K], \dots, a[K+m-1],$$

$$b[j] \in \{0,1\};$$

$$-K \leq i \leq K+m-1;$$

$$-K \leq j \leq K+n-1;$$

$$a[l]=b[r]=1;$$

$$h[l, r]=0$$

$$b[-K], \dots, b[K+n-1])) \wedge$$

$$\begin{aligned}
& (\wedge_{a[i] \in \{0,1\};} \neg z[a[-K], \dots, a[K+m-1], \\
& \quad b[j] \in \{0,1\}; \\
& \quad -K \leq i \leq K+m-1; \\
& \quad -K \leq j \leq K+n-1; \\
& \quad \exists l, r (l \neq r \wedge a[l] = a[r] = 1) \\
& \quad b[-K], \dots, b[K+n-1])) \wedge
\end{aligned}$$

$$\begin{aligned}
& (\wedge_{a[i] \in \{0,1\};} \neg z[a[-K], \dots, a[K+m-1], \\
& \quad b[j] \in \{0,1\}; \\
& \quad -K \leq i \leq K+m-1; \\
& \quad -K \leq j \leq K+n-1; \\
& \quad \exists l, r (l \neq r \wedge b[l] = b[r] = 1) \\
& \quad b[-K], \dots, b[K+n-1])) \wedge
\end{aligned}$$

$$\begin{aligned}
& (\wedge_{a[i] \in \{0,1\};} \neg z[a[-K], \dots, a[K+m-1], \\
& \quad b[j] \in \{0,1\}; \\
& \quad -K \leq i \leq K+m-1; \\
& \quad -K \leq j \leq K+n-1; \\
& \quad \forall r (a[r] = 0) \\
& \quad b[-K], \dots, b[K+n-1])) \wedge
\end{aligned}$$

$$\begin{aligned}
& (\wedge_{a[i] \in \{0,1\};} \neg z[a[-K], \dots, a[K+m-1], \\
& \quad b[j] \in \{0,1\}; \\
& \quad -K \leq i \leq K+m-1; \\
& \quad -K \leq j \leq K+n-1; \\
& \quad \forall r (b[r] = 0) \\
& \quad b[-K], \dots, b[K+n-1])) \wedge
\end{aligned}$$

$$\begin{aligned}
& (\wedge_{1 \leq q \leq K+1; u[p, q] \wedge v[r, q] \rightarrow} \\
& \quad 1 \leq p \leq m; \\
& \quad 1 \leq r \leq n; \\
& \quad a[i] = 0, i \neq p; \\
& \quad a[p] = 1, \\
& \quad b[j] = 0, j \neq r; \\
& \quad b[r] = 1
\end{aligned}$$

$$z[a[-K], \dots, a[K+m-1], \dots, b[K+n-1]] \wedge$$

$$(\wedge_{q=1}^K \wedge_{p=1}^m (x[1, q] \wedge u[p, q]) \rightarrow u[p-1, q+1]) \wedge$$

$$(\wedge_{q=1}^K \wedge_{p=1}^m (x[2, q] \wedge u[p, q]) \rightarrow u[p+1, q+1]) \wedge$$

$$(\wedge_{q=1}^K \wedge_{p=1}^m (\neg x[1, q] \wedge \neg x[2, q]) \wedge u[p, q]) \rightarrow u[p, q+1] \wedge$$

$$(\wedge_{q=1}^K \wedge_{p=1}^n (y[1, q] \wedge v[p, q]) \rightarrow v[p-1, q+1]) \wedge$$

$$(\wedge_{q=1}^K \wedge_{p=1}^n (y[2, q] \wedge v[p, q]) \rightarrow v[p+1, q+1]) \wedge$$

$$(\wedge_{q=1}^K \wedge_{p=1}^n (\neg y[1, q] \wedge \neg y[2, q]) \wedge v[p, q]) \rightarrow v[p, q+1] \wedge$$

$$(\wedge_{d=0}^{m-1} s[d, d, 1]) \wedge$$

$$(\wedge_{e=0}^{n-1} t[e, e, 1]) \wedge$$

$$(\wedge_{d=0}^{m-1} \wedge_{q=1}^{K+1} (\vee_{p=1}^m s[d, p, q])) \wedge$$

$$(\wedge_{d=0}^{m-1} \wedge_{q=1}^{K+1} (\vee_{\substack{p[1] \neq p[2] \\ 1 \leq p[1] \leq m; \\ 1 \leq p[2] \leq m}} \neg s[d, p[1], q] \vee \\ \neg s[d, p[2], q]))) \wedge$$

$$(\wedge_{e=0}^{n-1} \wedge_{q=1}^{K+1} (\vee_{p=1}^n t[e, p, q])) \wedge$$

$$(\wedge_{e=0}^{n-1} \wedge_{q=1}^{K+1} (\vee_{\substack{p[1] \neq p[2] \\ 1 \leq p[1] \leq n; \\ 1 \leq p[2] \leq n}} \neg t[e, p[1], q] \vee \neg t[e, p[2], q]))) \wedge$$

$$(\wedge_{d=1}^m \wedge_{e=1}^n ((\wedge_{q=1}^K \wedge_{p=1}^m (x[1, q] \wedge s[d, p, q]) \rightarrow s[d, p-1, q+1]) \wedge$$

$$(\wedge_{q=1}^K \wedge_{p=1}^m (x[2, q] \wedge s[d, p, q]) \rightarrow s[d, p+1, q+1]) \wedge$$

$$(\wedge_{q=1}^K \wedge_{p=1}^m (\neg x[1, q] \wedge \neg x[2, q] \wedge s[d, p, q]) \rightarrow s[d, p, q+1]) \wedge$$

$$(\wedge_{q=1}^K \wedge_{p=1}^n (y[1, q] \wedge t[e, p, q]) \rightarrow t[e, p-1, q-1]) \wedge$$

$$(\wedge_{q=1}^K \wedge_{p=1}^n (y[2, q] \wedge t[e, p, q]) \rightarrow t[e, p+1, q+1]) \wedge$$

$$(\wedge_{q=1}^K \wedge_{p=1}^n (\neg y[1, q] \wedge \neg y[2, q] \wedge t[e, p, q]) \rightarrow t[e, p, q+1])) \wedge$$

$$((\wedge_{p=1}^m ((s[d, p, 1] \vee \neg u[p, 1]) \wedge$$

$$(\neg s[d, p, 1] \vee u[p, 1]))) \vee ((\wedge_{p=1}^m ((t[e, p, 1] \vee \neg v[p, 1]) \wedge$$

$$(\neg t[e, p, 1] \vee v[p, 1]))) \vee$$

$$(\wedge_{1 \leq q \leq K+1; s[d, p, q] \wedge t[e, r, q] \rightarrow$$

$$1 \leq p \leq m;$$

$$1 \leq r \leq n;$$

$$a[i]=0, i \neq p;$$

$$a[p]=1;$$

$$b[j]=0, j \neq r;$$

$$b[r]=1$$

$$\neg z[a[-K], \dots, a[K+m-1], \dots, b[K+n-1]]))).$$

It is easy to check that this function gives us a SAT-encoding of VDLPP. Using standard transformations (see e.g. [3]), we can obtain an explicit transformation of SAT-encoding into 3SAT-encoding.

To solve VDLPP we use fgrasp and posit (see [4]). Also, we consider

A1 GSAT with adaptive score function (see [5]);

A2 genetic algorithm with exons and introns (see [6]);

A3 genetic algorithm with expansion operator (see [7]).

We use heterogeneous cluster (500 calculation nodes, Intel Core i7). We use the generator of natural instances for VDLPP (see [2]). We consider instances with m and n from 400 to 800. Selected experimental results for VDLPP are given in Tables 1, 2.

time	average	max	best
fgrasp	1.78 h	15.52 h	11.8 min
posit	3.22 h	12.4 h	4.55 min
A1	10.37 min	21.2 min	0.41 sec
A2	9.4 sec	7.4 min	2.06 sec
A3	13.7 sec	16.2 sec	11.8 sec

Table 1: Experimental results for explicit polynomial reduction from VDLPP to 3SAT where $n = m = 400$

time	average	max	best
fgrasp	3.24 h	26.73 h	18.9 min
posit	4.61 h	19.57 h	9.44 min
A1	1.33 h	3.72 h	5.3 sec
A2	11.25 min	56.43 min	2.81 min
A3	22.5 min	37.8 min	3.4 sec

Table 2: Experimental results for explicit polynomial reduction from VDLPP to 3SAT where $n = m = 800$

References

- [1] C. Tovey and S. Koenig, Gridworlds as Testbeds for Planning with Incomplete Information, *Proceedings of the AAAI Conference on Artificial Intelligence*, (2000), 819-824.
- [2] A. Gorbenko, V. Popov, and A. Sheka, Localization on Discrete Grid Graphs, *Lecture Notes in Electrical Engineering*, 107 (2012), 971-978.
- [3] A. Gorbenko and V. Popov, Task-resource Scheduling Problem, *International Journal of Automation and Computing*, 9 (2012), 429-441.
- [4] URL: <http://people.cs.ubc.ca/~hoos/SATLIB/>
- [5] V. Popov, GSAT with Adaptive Score Function, *Advanced Studies in Theoretical Physics*, 7 (2013), 363-366.
- [6] V. Popov, Genetic Algorithms with Exons and Introns for the Satisfiability Problem, *Advanced Studies in Theoretical Physics*, 7 (2013), 355-358.
- [7] V. Popov, A Genetic Algorithm with Expansion Operator for the 3-Satisfiability Problem, *Advanced Studies in Theoretical Physics*, 7 (2013), 359-361.

Received: June 15, 2013